## **Help Page**

## RepWords 1.1

RepWords 1.1 program is an implementation of the generalized Ruzzo-Tompa algorithm adjusted to detection of repeats in biological sequences. The program can be downloaded from the URL:

http://www.ncbi.nlm.nih.gov/CBBresearch/Spouge/html\_ncbi/html/index/software.html#18

Instructions for the installation can be found here.

## Usage.

The program is run with parameters separated by spaces. Each parameter of the command line is the pair:

-<parameter name> <parameter value>

#### Parameters.

The program can be executed in the two different modes:

- mode "RT": Ruzzo-Tompa mode. In this mode the program calculates repeats with affine gap penalties using the generalized Ruzzo-Tompa algorithm;
- mode "test": the test mode. In this mode the program runs a speed test for comparison of the two algorithms: "Generalized Ruzzo-Tompa algorithm" and "Divide-and-conquer algorithm".

Which mode is running is determined by the parameter "-mode" that is a required parameter.

The opening gap penalty d1 and the extending gap penalty d2 (defined in Ruzzo-Tompa mode of the program) assume the following convention: a gap of length k is penalized as d1+d2\*k.

## Mode "RT" (Ruzzo-Tompa mode)

## -mode <a mode of the program>

- Defines the mode and must have the value "RT".
- The parameter is required.

## -input\_w <a name of an input file with w-values>

- Each line of the "-input\_w" file has the format:w <string>
- The "-input\_w" file contains a list of w-values. Ruzzo-Tompa algorithm is applied for each w from the list.
- <string> is an arbitrary string ignored by the program in the current version.
- The parameter is optional (the parameter -w\_max can be used instead).

# -w\_max <maximum word-length w\_max>

- the calculation will be performed for each w from the interval [1, $w_m$ ].
- The parameter is optional (the parameter -input\_w can be used instead).

One of the parameters -input\_w or -w\_max is required but these parameters cannot be used together.

## -FASTA\_input <a name of an input file with a sequence in FASTA format>

- Only one sequence is permitted in the file.
- The parameter is required.

# -FASTA\_output <a name of a text output file with combined results>

• The parameter is optional but at least one of the parameters -FASTA\_output or -summary is required.

• If the parameter -summary is not defined, the score threshold used for the calculation is 1 (repeats with any scores are outputted). Otherwise, the minimum score threshold (from the file defined by -summary) is used for the calculation.

## -summary <a name of an input file with score thresholds and output files>

- Each line of the "-summary" file has the format: score threshold <file name>
- The program generated a summary file with the name <file name> for each score threshold.
- For each score threshold the program outputs only repeats with scores no less than the score threshold.
- Each file contains a full list of repeats with corresponded w.
- The parameter is optional but at least one of the parameters -FASTA\_output or -summary is required.

The parameters -FASTA\_output or -summary can be used together.

## -gap\_open <the opening gap penalty>

- Must be a non-negative integer number.
- The parameter is ignored in the case of the gapless repeats (when "-gapped false" is defined).
- The parameter is optional (the default value is 1).

## -gap\_extend <the extending gap penalty>

- Must be a positive integer number.
- The parameter is ignored in the case of the gapless repeats (when "-gapped false" is defined).
- The parameter is optional (the default value is 1).

# -scoring\_matrix <a name of an input file with the scoring matrix>

• The format of the file as follows. The first line contains a positive integer number **B** of letters in the alphabet. The rest of the file is a **B**x**B** table with **B** rows and **B** columns. The element from the row **a** and the column **b** of the

- table is an integer number corresponded to the similarity score between the letters with the order numbers **a** and **b**.
- The parameter is optional (the default value corresponds to a four letters alphabet with the matrix 2/-3 (diagonal elements of the matrix equal 2, all other elements equal -3)).

## -alphabet\_yes <a name of an input file with a list of valid letters>

- The first line of the file contains a number of letters in the alphabet; the second line contains the letters in a fixed order.
- The order of the letters corresponds to the order of columns and rows of the scoring matrix.
- The parameter is optional (the default alphabet is "ACTG").

The parameters -gap\_open, -gap\_extend, -scoring\_matrix, -alphabet\_yes must be defined or omitted simultaneously.

## -gapped <gap penalty flag>

- Defines whether the repeats are gapped or not; the setting "-gapped true" corresponds to gapped repeats.
- The setting "-gapped false" corresponds to the gapless case and the parameters "-gap\_open" and "-gap\_extend" are ignored in this case.
- The parameter is optional (the default value is "-gapped true").

# -sequences\_number <number of parts>

- Defines a number of parts the input sequence is split into during the calculation. The result is combined from the results for the individual parts. It gives a possibility to process larger input sequences with more efficient memory use. The larger number of parts, the smaller amount of memory is required.
- The parameter is optional (the default value is 1).

# -prerepeat\_is\_separated<X-letters flag>

• Let's consider the following repeat for **w**=21: AACTTTGTXXXXXXXXXXXXXXAACTTTGT

- Here there is a match between 2 identical words "AACTTTGT".
- The parameter defines whether the X-letters ("XXXXXXXXXXXXX" in the example) is a part of repeats (value "false") or not (value "true").
- Optional parameter (the default value is "true")

## -output\_line\_size <length of lines of the output files>

- Can be set to some large number like 1000000000 to ensure one-line output.
- The parameter is optional (the default value is 70).

## -w\_choice <method for w assignment>

- Defines the way of how w is assigned to a repeat.
- Let's some position of the input sequence is included into several overlapping repeats corresponded to different w.
- The parameter equals "w\_min" if a minimum w is selected for the position among all w of the overlapping repeats.
- The parameter equals "score" if w, selected for the position, corresponds to the repeat with a maximum score among the overlapping repeats.
- In the case when summary files (please see the parameter -summary) are generated for different score thresholds, the program guaranties exact summary results for all score thresholds in the case "-w\_choice score". In the case "-w\_choice w\_min", the program guaranties exact summary results only for the minimum score threshold (the calculation should be repeated for each score threshold separately in this case).
- The parameter is optional (the default value is "w\_min").

# -HTML\_output <a name of an output HTML file with combined results>

- The parameter is optional (the program does not produce an HTML output if the parameter is missing).
- The score threshold used for the calculation is the same as for the text output defined by -FASTA\_output.

## -input\_colors <a name of an input file with colors>

• The colors are used in the HTML output; the file explains how to color repeats corresponded to different w.

- Each line has the format:
  - <w> <RGB hexadecimal code of the color>
- Can be set only if the parameter "-HTML\_output" is defined.
- If a color is not defined for some w listed in the file "-input\_w", then the missing color is assigned to the color corresponded to the maximum w defined by the parameter -input\_colors.
- The parameter is optional (the program uses default colors for each w if the parameter is missing).

-explain\_colors <a name of an output file with explanations of colors for different **w**>

- Can be set only if the parameter "-HTML output" is defined.
- The parameter is optional (the file is not outputted if the parameter is missing).

#### Mode "test" (the test mode)

## -mode <a mode of the program>

- Defines the mode and must have the value "test".
- The parameter is required.

# -sequence\_length <length of test sequences>

- Must be a positive integer number.
- The parameter is required.

# -gap\_open <the opening gap penalty>

- Must be a non-negative integer number.
- The parameter is optional (the default value is 1).

# -gap\_extend <the extending gap penalty>

- Must be a positive integer number.
- The parameter is optional (the default value is 1).

## -scoring\_matrix <a name of an input file with the scoring matrix>

- The parameter is optional.
- Please see the description of -scoring\_matrix parameter for the mode "RT".

## -frequencies\_input <a name of an input file with the background frequencies>

- The format of the file as follows. The first line contains a positive integer number **B** of letters in the alphabet. The next **B** lines contain the background frequencies: one real number per each line. The sum of the background frequencies must be equal to 1.
- The parameter is optional (the default value is four letters alphabet with the frequencies (0.25,0.25,0.25,0.25)).

The parameters -gap\_open, -gap\_extend, -scoring\_matrix, -frequencies\_input must be defined or omitted simultaneously.

## -w <the word-length>

- Must be a positive integer number.
- The parameter defines the word-length **w** used in the calculation.
- The parameter is optional (the default value is 1).

## -sequences\_number <a number of test sequences>

- Must be a positive integer number.
- The parameter is optional (the default value is 1).

#### -trials <a number of trials>

- Must be a positive integer number.
- The first trial uses the input sequence length (defined by the parameter "-sequence\_length"). The sequence length of the **K**th trial is the input sequence length multiplied by **2**^(**K-1**).
- For example, in the case "-sequence\_length 128 -sequences\_number 100 trials 8", the program performs 8 different tests for the lengths 128, 256, 512, 1024, 2048, 4096, 8192, 16384 and each test generates 100 random sequences of the corresponding length (incremented by w) according to the background frequencies.

• The parameter is optional (the default value is 1).

## -screen\_output <screen output flag>

- Determines whether the program outputs the resulted maximal intervals (repeats) on the screen (the value "true") or not (the value "false").
- The parameter is optional (the default value is "false").

#### -srand <the randomization number>

- Must be a non-negative integer number.
- Defines a seed for pseudorandom numbers generated in the program.
- If the parameter equals 0, then the randomization number is generated inside the program (from the system time).
- The randomization number is outputted on the screen.
- The program exactly reproduces the output if the same randomization number and other parameters are used.
- The parameter is optional (the default value is 0).

## Output.

The program outputs some information on the screen and into files depending on the parameters.

## Screen output.

## Screen output in mode "RT":

• The program displays general progress of the calculation.

# Screen output in mode "test":

- The randomization number.
- Calculation progress.

- Calculation times for each method for different sequence lengths.
- Calculated maximal intervals in the case when the parameter "screen\_output" is "true".

## File output.

## File output in mode "RT".

The program can generate several files depending on the parameters:

- The program outputs a sequence in the text format with repeats masked if the parameter **-FASTA\_output** is defined. The repeats are masked by low case.
- The program outputs a list of all repeats and corresponded **w** if the parameter -summary is defined; there is a possibility to generate repeats for different score thresholds in a single run.
- The program outputs masked sequence in the HTML format if the parameter
   -HTML\_output is defined. The repeats are masked by low case and different
   w are masked by different colors. The parameter -input\_colors defines what
   color should be used for different w. The file can be viewed by any Internet
   browser.
- The program outputs an HTML file with explanations of colors if the
  parameter -explain\_colors is defined. The colors are used in the file defined
  by the parameter -HTML\_output to mask repeats corresponded to different
  w.

# File output in mode "test".

No file output generated in the test mode.

# **Examples of the command line.**

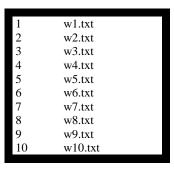
#### Mode "RT"

-mode RT -FASTA\_input seq.in -FASTA\_output seq1.out -input\_w w10.in - gap\_open 1 -gap\_extend 1 -scoring\_matrix matr4.in -alphabet\_yes alphabet\_ACGT.in -sequences\_number 3 -gapped true -summary summary.in - prerepeat\_is\_separated true -output\_line\_size 70 -w\_choice score -HTML\_output seq1.html -explain\_colors colors01\_10.html -input\_colors colors01\_10.in

-mode RT: defines the Ruzzo-Tompa mode.

-FASTA\_input seq.in: defines an input file with a sequence in FASTA format:

- -FASTA\_output seq1.out: name of an output text file;
- -input\_w w10.in: defines a file with input w:



- -gap\_open 1 -gap\_extend 1: the affine gap penalty is 1/1.
- -scoring\_matrix matr4.in: defines a name of a file with a scoring matrix:

4				
2	-3	-3	-3	
-3	2	-3	-3	
4 2 -3 -3	-3	2	-3	
-3	-3	-3	2	

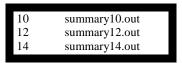
-alphabet\_yes alphabet\_ACGT.in: defines a file with the alphabet:



-sequences\_number 3: the input sequence is split into 3 parts.

-gapped true: the calculation is gapped.

-summary summary.in: defines a file with score thresholds and corresponded file names for the output:



The program outputs 3 summary files for the score thresholds 10, 12 and 14.

-prerepeat\_is\_separated true: X-letters are excluded from a repeat.

-output\_line\_size 70: a line of an output file has length 70.

-w\_choice score: defines the choice of **w** for positions of the sequence.

-HTML\_output seq1.html: defines an output file in the HTML format.

-explain\_colors colors01\_10.html: defines an output file with explanations of colors used in the HTML output file with repeats.

-input\_colors colors01\_10.in: defines a file with colors used for different **w** in the HTML output file:

```
1 0xFF0000
2 0x00DD00
3 0x0000DD
4 0xFFD700
5 0x00DDDD
6 0x800000
7 0x008000
8 0x000080
9 0x008080
10 0xFF00FF
```

The following example of the command line does not define all parameters (default values are used instead):

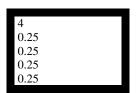
-mode RT -FASTA\_input seq.in -FASTA\_output seq1.out -w\_max 10

The score threshold is 1 in this example.

#### Mode "test"

-mode test -w 16 -gap\_open 1 -gap\_extend 1 -scoring\_matrix matr4.in - frequencies\_input RR4.in -sequence\_length 128 -sequences\_number 100 -trials 8 - screen\_output false -srand 76832338

The program performs tests for 8 different sequence lengths 128, 256, 512, 1024, 2048, 4096, 8192, 16384 and for each test the program generates 100 random sequences of the corresponding length (incremented by  $\mathbf{w=16}$ ) according to the background frequencies defined in the file "RR4.in":



The program uses the randomization seed 76832338 and does not output resulted maximal intervals on the screen. The affine gap penalties are 1/1 (a gap of length k is penalized as 1+k); the scoring matrix is extracted from the file "matr4.in".

The program outputs the calculation times for each sequence length and method on the screen.

#### Files and Installation

The files in the download directory include:

- 1. repwords\_1.1\_WINDOWS.zip: Windows executable.
- 2. repwords\_1.1\_LINUX.zip: LINUX executable.
- 3. repwords\_1.1\_cpp\_files.zip: C++ source files.
- 4. repwords\_1.1\_examples\_files.zip: contains the following sample files:
  - w10.in: an example of an input file for the parameter "-input\_w".
  - matr4.in: an example of an input file with a scoring matrix.
  - RR4.in: an example of an input file with background frequencies.
  - alphabet\_ACGT.in: an example of an input file with alphabet letters.
  - seq.in: an example of an input file with a sequence in FASTA format.
  - summary.in: an example of an input file with score threshold and corresponded summary file names.
  - mode\_RT.bat, mode\_test.bat: Windows batch files to run examples in modes "RT" and "test" respectively.
  - No special installation is required.
  - The executable files can be downloaded, unzipped and run with the appropriate command line.
  - Alternatively, the source C++ files can be downloaded, unzipped, and complied in a suitable C++ environment.

#### Remark.

To compile the C++ files under UNIX, please replace the line #define \_MSDOS\_ by the line

//#define \_MSDOS\_ in the file "sls\_repwords.h".